Computers and Mathematics with Applications 62 (2011) 2685–2695

Contents lists available at SciVerse ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

An evolutionary particle filter with the immune genetic algorithm for intelligent video target tracking

Hua Han, Yong-Sheng Ding*, Kuang-Rong Hao, Xiao Liang

College of Information Sciences and Technology, Donghua University, Shanghai 201620, PR China
Engineering Research Center of Digitized Textile & Fashion Technology, Ministry of Education, Donghua University, Shanghai 201620, PR China

ARTICLE INFO

Keywords:
Target tracking
Intelligent surveillance
Particle filter
Immune genetic algorithm
Sample impoverishment
Re-sampling

ABSTRACT

Particle filter algorithm is widely used for target tracking using video sequences, which is of great importance for intelligent surveillance applications. However, there is still much room for improvement, e.g. the so-called "sample impoverishment". It is brought by re-sampling which aims to avoid particle degradation, and thus becomes the inherent shortcoming of the particle filter. In order to solve the problem of sample impoverishment, increase the number of meaningful particles and ensure the diversity of the particle set, an evolutionary particle filter with the immune genetic algorithm (IGA) for target tracking is proposed by adding IGA in front of the re-sampling process to increase particle diversity. Particles are regarded as the antibodies of the immune system, and the state of target being tracked is regarded as the external invading antigen. With the crossover and mutation process, the immune system produces a large number of new antibodies (particles), and thus the new particles can better approximate the true state by exploiting new areas. Regulatory mechanisms of antibodies, such as promotion and suppression, ensure the diversity of the particle set. In the proposed algorithm, the particle set optimized by IGA can better express the true state of the target, and the number of meaningful particles can be increased significantly. The effectiveness and robustness of the proposed particle filter are verified by target tracking experiments. Simulation results show that the proposed particle filter is better than the standard one in particle diversity and efficiency. The proposed algorithm can easily be extended to multiple objects tracking problems with occlusions.

1. Introduction

There has been an increasing interest in target tracking of video sequences, which is particularly important to intelligent surveillance applications in recent years [1–5]. The issue of continuous video tracking based on the image sequences of interested regions has aroused people's extensive attention. The current difficulties for video tracking lie in the following: (1) Due to the disturbance of complex background, there is a shortage of effective means for extracting target motion areas; (2) Since the target is obscured by obstacles, intermittent phenomenon appears in measurement data; (3) The issue of cross-trajectory tracking caused by dense targets. In order to solve the above problems, varieties of algorithms have been proposed in papers [6–9]. The particle filter is one of the most widely used algorithms mentioned above, especially in the scope where the problem reduces to the reconstruction of the probability density function of the target state given measurements and prior knowledge [10]. The particle filter is also known as Condensation [11], Bootstrap Filter [12], and Sequential Monte

* Corresponding address: College of Information Sciences and Technology, Donghua University (Former: China Textile University), Shanghai 201620, PR China. Tel.: +86 21 67792329; fax: +86 21 67792353.
E-mail addresses: ysding@dhu.edu.cn (Y.-S. Ding), krhao@dhu.edu.cn (K.-R. Hao).

0898-1221/$ – see front matter
doi:10.1016/j.camwa.2011.06.050

Carlo Filter [13] proposed in signal processing, computer vision, statistics, respectively, and other areas to resolve non-Gaussian, nonlinear Bayesian recursive filtering problem. Although the particle filter is widely used, room for improvement still exists. In visual tracking, there are three main factors that affect the performance of the particle filter, namely, the sample impoverishment, the reliable observation model, and the accurate motion model, which are usually regarded as key research points.

Most researchers who are interested in the performance of the particle filter make great effort on constructing reliable observation model [14–16] and accurate motion model [3,17,18]. Maggio and Cavallaro [14] used semi-overlapping color histograms to improve the sensitivity to rotations and anisotropic scale changes, which could increase the efficiency of the particle filter. This method was well performed in general situation. But the authors did not discuss the cases with occlusion which often happened in video surveillance. Kristan et al. [3] tried to construct a two-stage dynamic model which included a liberal model and a conservative model. The proposed method allowed larger perturbations in the target's dynamics. Due to the two-stage dynamic model's ability to actively adapt to the target's motion during tracking, it improved the performance of the particle filter. But all of the above-mentioned methods have not solved the inherent shortcomings of the particle filter itself, i.e., sample impoverishment.

Sample impoverishment is brought by re-sampling which is introduced to avoid particle degradation. When impoverishment occurred, the problem would seriously affect the particle filter's description ability of moving target's state. Some researchers have studied this problem. For example, Park et al. [19] proposed a new evolutionary particle filter to prevent sample impoverishment. They exploited the advantages of the evolutionary algorithm in the particle filter. They rigorously account the change of the target distribution caused by the genetic operators such as crossover and mutation. But their work is only at a theoretical proof, and not used in practical applications.

In this paper, an evolutionary particle filter with the immune genetic algorithm (IGA) for target tracking is proposed. The main contributions of this study lie in the follows. Regard the target's state and particles as the foreign invasion antigens and antibodies respectively. We use the crossover, mutation, and other operations to generate new particles, and use the immune regulation mechanism, memory mechanism, and others to maintain the diversity of particles. All of these operations can improve the meaningful particle number and reduce sample impoverishment. As such, the effectiveness and accuracy of target tracking can be improved.

The rest of the paper is organized as follows. Section 2 presents the particle filter for video target tracking, and describes the motion model, target representation, and likelihood model. Section 3 presents an evolutionary particle filter with IGA, includes the design and evaluation of the new proposed algorithm. Simulation and the related results of the proposed particle filter with IGA are provided in Section 4. Finally, Section 5 presents the conclusion of the whole paper.

2. A particle filter for video target tracking

2.1. Particle filter algorithm

Particle filter algorithm is a new nonlinear and non-Gaussian filtering method based on sequential importance sampling. It refers to an estimation process, that is, by means of finding a group of random samples that propagate in state space to approximate the probability density function, making sample mean instead of integral operations to gain the state minimum variance. And these samples are referred to as "particles". As the number of particles increases, the particles' probability density function gradually approaches the state probability density function, and the effect of the particle filter estimation reaches an optimal Bayesian estimation [20].

Generally, the dynamic time-varying systems are described as follows:

$$\begin{aligned} X_k &= f_k\left(X_{k-1}, U_k\right) \\ Z_k &= h_k\left(X_k, V_k\right), \end{aligned} \tag{1}$$

where $X_k \in R^{n_x}$ denotes the system state at time step k, $Z_k \in R^{n_z}$ denotes the measurement sequence which are related to the state vector via the observation equation. U_k and V_k are independent and identically distributed system noise and observation noise, respectively. In the Bayes sense, numerical solution needs high-dimensional integral for many nonlinear problems. The Monte Carlo method based on random sampling computation can convert integral into limited samples' summation, that is, the state probability density distribution can be approximately expressed by empirical probability distribution [21]. Suppose that the state is a first-order Markov process, state X_k and measurement sequence Z_k are independent of each other, and the initial state's (X_0) prior distribution is $p\left(X_0\right)$. Taking N independent and identically distributed samples $\left\{X_{0:k}^i;\ i = 1, 2, \ldots, N\right\}$ from $p\left(X_{0:k}|Z_{1:k}\right)$, the state posterior probability density can be approximated as follows by the empirical distribution [19]

$$p(X_{0:k}|Z_{1:k}) \approx \frac{1}{N}\sum_{i=1}^{N}\delta\left(X_{0:k} - X_{0:k}^{(i)}\right), \tag{2}$$

where $\delta\left(\cdot\right)$ denotes the Dirac δ function. In practice, however, $p(X_k|Z_{1:k})$ may be multi-variable, non-standard probability distribution, so it is usually impossible to sample directly from the probability density function of state, but needs some sampling algorithms [22]. The importance function is a distribution function that the probability distribution is the same as $p(X_k|Z_{1:k})$, and the probability density distribution $q(X_{0:k}|Z_{1:k})$ is known and easily to be sampled. Taking N samples $\{X^i_{0:k}; i = 1, 2, \ldots, N\}$ form the importance distribution function $q\,(X_{0:k}|Z_{1:k})$, the probability density function of state can be approximated by the weighted sum [21],

$$\begin{aligned} p(X_{0:k}|Z_{1:k}) &\approx \sum_{i=1}^{N} \varpi^{(i)} \delta(X_{0:k} - X^{(i)}_{0:k}) \\ \varpi^{(i)} &= \omega^{(i)} \Big/ \sum_{i=1}^{N} \omega^{(i)}, \end{aligned} \tag{3}$$

where $\omega_k(X_{0:k}) = \frac{p(Z_{1:k}|X_{0:k})p(X_{0:k})}{q(X_{0:k}|Z_{1:k})}$ is called importance weights. The criterion of selecting importance function is to minimize the variance of the importance weight. In the condition that $X^{(i)}_{k-1}$ and $Z_{0:k}$ are known, the optimal importance function is $q\left(X_k|X^{(i)}_{0:k-1}, Z_{1:k}\right) = p\left(X_k|X^{(i)}_{0:k-1}, Z_k\right)$, which is called optimal importance distribution function with the corresponding importance weight $\omega^{(i)}_k = \omega^{(i)}_{k-1} p\left(Z_k|X^{(i)}_{k-1}\right)$. But the drawback is that apart from some linear Gaussian models, the integral usually has no analytical solution. For application, most importance functions are carried out on sub-optimal algorithm. And many researchers try to generate better importance distribution [23], but it still not an optimal one. So a particle filter is just an approximation to the analytical filter. A standard particle filter consists of the following steps as shown in Algorithm 1.

Algorithm 1. The standard particle filter algorithm	
Step 1. *Initialization*	$k = 0$, select reference visual target manually in the initial frame, establish initial particle set $\left\{X^{(i)}_0, \omega^{(i)}_0\right\}_{i=1}^{N}$ according to state prior distribution $p\,(X_0)$, where $\omega^{(i)}_0 = \frac{1}{N}$
Step 2. *Particle state transition*	$k = 1, 2, 3, \ldots$,calculatethenewparticleset$\left\{\tilde{X}^{(i)}_k\right\}_{i=1}^{N}$accordingto the random drift model and the particle set $X^{(i)}_{k-1}$
Step 3. *Calculation of particle weight*	Calculate according to Eq. (3) and normalize to obtain $\omega^{(i)}_k$
Step 4. *State estimation output*	Calculate the MMSE Estimation $\hat{X}_k = E\,(X_k) = \sum_{i=1}^{N} \omega^{(i)}_k \tilde{X}^{(i)}_k$ of the target state at k time step
Step 5. *Re-sampling*	Re-sample N particles from $\left\{\tilde{X}^{(i)}_k, \omega^{(i)}_k\right\}_{i=1}^{N}$ according to $\omega^{(i)}_k$, and obtain a new particle set $\left\{X^{(i)}_k, \omega^{(i)}_k\right\}_{i=1}^{N}$, where $\omega^{(i)}_k = 1/N$
Step 6. *Cycles*	Go to *Step* 2

2.2. Motion model

The interested reference target, X_c, is chosen manually. In the case mentioned in this paper, X_c is an ellipse region. We track the ellipse's center (more detailed description of the target region model can be seen in Section 2.3). The motion of the moving target is modeled by the random walk model. In the sense of random walk model, the position of visual target in frame k consists of position in frame $k-1$ and the Gaussian noise disturbance. Suppose the state vector X of visual target expresses its position on two-dimensional image domain, and the state of time $k-1$ is X_{k-1}, so the state of time k is

$$X_k = X_{k-1} + U_k, \tag{4}$$

where U_k is the white Gaussian noise. In order to gain better tracking performance, we generally assume that the random component is large relatively (i.e., large noise variance).

2.3. Target representation and likelihood model

In this paper, the target feature is approximated as an ellipse and its visual target state is represented by $X = (Y, S)$ in which $Y = [x, y]$ is the center of the ellipse and meanwhile $S = [h_1, h_2, \theta]$. The variable h_1 is the length of minor semi-axis, h_2 is the length of major semi-axis, θ is the ellipse eccentric angle. Such relationships are shown in Fig. 1.

For the video object, under the assumption of first-order Markov process, observation probability $p(Z_k|X_k)$ is defined as a probability distribution of some kind of visual features. Therefore, the calculation of $p(Z_k|X_k)$ depends on the statistical description of the targets. Here we describe the color feature based on kernel and establish the observation probability distribution of visual target [24,25]. The weighted color distribution of pixels inside the ellipse represents the target and is

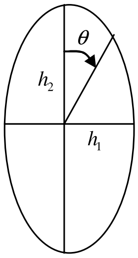


Fig. 1. Parameter definition of the ellipse bounding the target feature.

approximated by a normalized color histogram. Suppose the video object's color distribution is separated into B-class (RGB space normally taken $B = 8 \times 8 \times 8$), and color quantization function $b(l_m) : R^2 \to \{1, \ldots, B\}$ is which express assigning the pixel's color value of location l_m to color distribution's corresponding class. Therefore, for video target state X, the color distribution is defined as [26]

$$p_l^{(u)} = C \sum_{m=1}^{M} k\left(\left\| \frac{l - l_m}{h} \right\|\right) \delta(b(l_m) - u), \tag{5}$$

where l is the center (x, y) of video target determined by target X, M is the total number of pixels in the target area and $h = \sqrt{h_x^2 + h_y^2}$ is the size of target area, $k(\cdot)$ is kernel function (usually select Gaussian kernel), $\delta(\cdot)$ is the Kronecker delta function and C is the normalization constant, and

$$C = \frac{1}{\sum_{m=1}^{M} k\left(\left\| \frac{l-l_m}{h} \right\|\right)}. \tag{6}$$

For the calculation of video target's observation probability, we select the reference target X_c in the initial frame and establish the reference target's color distribution $\{q^{(u)}\}_{u=1,\ldots,B}$. In kth frame, suppose the image area's corresponding color distribution of video target state X_k's ith sampling $X_k^{(i)}$ is $\{p^{(u)}\}_{u=1,\ldots,B}$. Sampling $X_k^{(i)}$ is an assumption state in kth frame. So, the similarity measure between the sample and reference target can be established by using color distribution. Here, the Bhattacharyya coefficient is applied to set up which is defined as follow [6,25]

$$\rho[p^{(u)}, q^{(u)}] = \sum_{u=1}^{B} \sqrt{p^{(u)} q^{(u)}}. \tag{7}$$

So the similarity measure between sample $X_k^{(i)}$ and reference target X_c can be defined as

$$D(p, q) = \sqrt{1 - \rho[p^{(u)}, q^{(u)}]}, \tag{8}$$

where $D(p, q)$ is the Bhattacharyya distance. Finally, the observation probability distribution can be defined as:

$$p(Z_k^{(i)} | X_k^{(i)}) = \frac{1}{2\pi} \mathrm{e}^{-\lambda D^2(p,q)/2}, \tag{9}$$

where λ is the controlling parameters. So weight $w_k^{(i)}$ of the particle set $\tilde{X}_k^{(i)}$ can be calculated according to Eq. (9). After being normalized, we can derive that $w_k^{(i)} = w_k^{(i)} / \sum_{i=1}^{N} w_k^{(i)}$.

2.4. Evaluation of the standard particle filter

In the standard particle filter, degradation occurs frequently which performs as follows. After several recursive calculations, except for a few particles, the remaining particles' weight is almost negligible so that a large number of recursive calculation is wasted on the updates of particles which almost does not work. Even there is only a large weighted (almost close to 1) effective particle eventually, and other particles' weight is almost zero, it still produces a degradation distribution [20]. This leads to a number of trajectories updated by the state that cannot afford any effect. It also wastes a lot of computing resources while reducing the performance of the particle filter [21]. Particle degradation phenomenon emerges from the limited sampling and limited computer word length. When there is a narrow support set and with very little overlap between the proposal distribution $q\,(X_k | X_{0:k-1}, Z_{1:k})$ and the posterior distribution $p(X_k | Z_{1:k})$, because of limited sampling, most of the particles are not on the public support sets. Therefore, after the Bayesian update of the measurement, the particles would have smaller weight. And repeated recursive calculations and computer underflow induce most of the particles' weight to be zero and become ineffective particles, which reduces the number of effective particles. The number of effective particles is defined as [19],

$$N_{\mathrm{eff}} = \frac{N}{1 + \mathrm{Var}_{q(\cdot|Z_{0:k})}\left(\tilde{\omega}_k^{(i)}\right)} = \frac{N}{E_{q(\cdot|Z_{0:k})}\left[\left(\tilde{\omega}_k^{(i)}\right)^2\right]}. \tag{10}$$

From Eq. (10) we can see that N_{eff} is difficult to calculate, so the number of effective particles can be approximated as

$$\hat{N}_{\mathrm{eff}} = \frac{1}{\sum_{i=1}^{N}\left(\omega_k^{(i)}\right)^2}. \tag{11}$$

Apparently, one or more particles cannot approximate the statistical characteristics (or probability distribution). Therefore, degradation would seriously affect the performance of the particle filter. In general, there are three methods to avoid degradation phenomena. The first is to increase the number of particles. However, methods to increase particle number would lead to computational complexity, which is the disadvantage of real-time tracking. The second recommendation is to select an effective proposal distribution. The classic sampling strategy includes rejection sampling, importance sampling, and Markov–Monte Carlo sampling. These sampling strategies can be directly applied in the particle filter. In recent years, the researchers conducted extensive research to improve the sampling methods. However, these improved methods are basically dependent on the three classical sampling strategies. Therefore, it is difficult to make a major breakthrough of sampling strategies. The third is re-sampling. The re-sampling method is to introduce re-sampling step after the weight calculation. Re-sampling, in other words, sample the particle set $\left\{\tilde{X}_k^{(i)}, \omega_k^{(i)}\right\}_{i=1}^{N}$ again according to particles' weight. Larger weighted particles are extracted repeatedly (sampling frequency is counted as N_i, and $\sum_{i=1}^{N} N_i = N$), while smaller weighted particles are randomly removed [20].

Although re-sampling can eliminate the smaller weighted particles' impact, it also introduces a new negative issue, which is known as sample impoverishment [20]. In particular, when there is fewer system noise, the sample impoverishment becomes more seriously. Sample impoverishment arises from the re-sampling process. Owing to the high weighted particles' over-replication, the quantity of meaningful particles reduced, which leads to the information capacity of new particle set seriously reduced. As a result, after several recursive calculation, effective particles are almost exhausted, and the new particle set is difficult to reflect the true statistical state properties. In this paper, we propose a new method by adding the immune genetic process before re-sampling to ensure the effectiveness and the diversity of the particle sets.

3. The evolutionary particle filter with the immune genetic algorithm

3.1. The immune genetic algorithm for particle filter

The IGA is a type of algorithm inspired by biological immune and genetic mechanism. It simulates the adaptive capacity of biological immune system which produces antibodies against the foreign antigens' invasion. It achieves the recognition of invasive antigens, the generation of diverse antibodies, self-regulation, immune memory and other functions [27]. The foreign invasion antigens and the antibodies generated by the immune system respectively correspond to the objective function of practical problems and the candidate solution, that is, the state of being tracked targets correspond to antigens, and particles correspond to antibodies. The immune optimization algorithm simulates the mechanism of antibody diversity keeping of the immune system. It achieves the self-regulation function, that is, uses the mechanism of antibody's concentration selection to achieve the promotion and inhibition of antibodies, which can maintain the diversity. The selection probability p_i of antibodies consists of probability of fitness p_{fi} and probability of concentration inhibition p_{di}, specifically as follows [27]

$$p_i = \alpha p_{\mathrm{fi}} + (1-\alpha)\, p_{\mathrm{di}} = \alpha \frac{F(i)}{\sum_{j=1}^{N} F(j)} + (1-\alpha)\, \frac{1}{N} \mathrm{e}^{-\frac{C_i}{\beta}}, \tag{12}$$

where α and β is adjustment constant, respectively, N is the total number of antibodies, C_i is the concentration (which is the proportion of similar antibodies) and $F(\cdot)$ is the fitness function. C_i is calculated as follows

$$C_i = \frac{n}{N}, \tag{13}$$

where n is the number of antibodies with high affinity of antibody i, N is the total number of all antibodies. The affinity degree between antibodies is evaluated by Euclidean distance. Suppose Ab_1 and Ab_2 are two different particles (antibodies), and

$$\text{dist} = \sqrt{\sum_{i=1}^{n}(Ab_1 - Ab_2)^2}, \tag{14}$$

where n is the number of attributes of Ab_1 and Ab_2. The maximum distance between any two vectors is defined as

$$\max\text{dist} = \sqrt{\sum_{i=1}^{n} r_i^2},$$

where r is the range of values for attribute i. So the affinity between particle (antibody) Ab_1 and Ab_2 is defined as follow

$$\text{affi}_{Ab_1,Ab_2} = \frac{\text{dist}}{\max\text{dist}}. \tag{15}$$

Calculation of fitness $F(\cdot)$: Consider the evolution can only be toward the direction of fitness function increasing, the fitness function can be constructed by the inverse of root mean square error between state estimate and true state, make the objective function $\min e_k = |X_k - \hat{X}_k|$, so it can be derived that

$$F(i) = 1/e_i. \tag{16}$$

Crossover and mutation [27]: Crossover can operate between two cross parents which are selected randomly according to the crossover probability p_c, parent produce offspring by linear cross means. Here the float-coding scheme is applied as

$$\begin{cases} x_1' = rx_1 + (1-r)x_2 \\ x_2' = (1-r)x_1 + rx_2, \end{cases} \tag{17}$$

where r is a random number generated from interval [0, 1]. Mutation can operate between two mutation parents which are selected randomly according to the mutation probability p_m. Parents produce offspring by non-uniform mutation operation, that is the variance of the mutation operation involved in the generation of antibodies is non-uniform change. So

$$x' = \begin{cases} x + \Delta[g_c, r(k) - x], & \text{sign} = 0 \\ x - \Delta[g_c, x - l(k)], & \text{sign} = 1 \end{cases} \tag{18}$$

$$\Delta(g_c, y) = yr\left(1 - \frac{g_c}{T}\right)^b$$

where Δ is function to compute element change, y is the distance between x and boundary values. g_c is the current generations of evolution, T is the maximum evolution generation and r is shape factor which can modulate non-uniform change of function carve.

The proposed particle filter with IGA is effective to solve the sample impoverishment brought by re-sampling. The proposed algorithm applies the immune optimization process before re-sampling, which makes full use of the mechanism of the immune system, such as the promotion and inhibition of antibody concentration, crossover, mutation, memory and others. This cannot only guarantee the high weighted particles still in the memory unit, but also regulate the concentration of antibodies. The high-frequency mutation and particle crossover can make the original particle set diffuse, and thus obtain new particles to improve the diversity of the particle set. The corresponding algorithmic process is shown in Algorithm 2.

Algorithm 2. The evolutionary particle filter with IGA	
Step 1. *Initialization*	$k = 0$, select reference visual target manually in the initial frame, establish initial particle set $\left\{X_0^{(i)}, \omega_0^{(i)}\right\}_{i=1}^{N}$ from $p(X_0)$, where $\omega_0^{(i)} = \frac{1}{N}$
Step 2. *Particle state transition*	$k = 1, 2, 3, \ldots$, calculatethenewparticleset $\left\{\tilde{X}_k^{(i)}\right\}_{i=1}^{N}$ according to the random drift model and particle set $X_{k-1}^{(i)}$
Step 3. *Calculation of particle weight*	Calculated according to Eq. (3) and normalized to obtain $\omega_k^{(i)}$
Step 4. *State estimation output*	Calculate the MMSE Estimation $\hat{X}_k = E(X_k) = \sum_{i=1}^{N} \omega_k^{(i)} \tilde{X}_k^{(i)}$ of the target state at k time step
Step 5. *Fitness calculation*	Do it according to Eq. (16)
Step 6. *Memory unit update*	Update the memory unit, and maintain the antibody's diversity at the same time
Step 7. *Concentration regulation*	Antibodies with high concentration should reduce their choosing probability, and vice versa

(*continued on next page*)

Step 8. *Crossover*	Do it according to Eq. (17)
Step 9. *Mutation*	Do it according to Eq. (18)
Step 10. *Cycles*	Cycles from *Step* 6 to *Step* 9. Cycle until the global error $e_k \leq \varepsilon$, make the memory unit as a new particle set
Step 11. *Re-sampling*	Re-sample N particles from $\left\{\tilde{X}_k^{(i)}, \omega_k^{(i)}\right\}_{i=1}^{N}$ according to $\omega_k^{(i)}$, and obtain a new particle set $\left\{X_k^{(i)}, \omega_k^{(i)}\right\}_{i=1}^{N}$, where $\omega_k^{(i)} = 1/N$
Step 12. *Cycles*	Make $k = k + 1$, goto *Step* 2

3.2. Evaluation of the evolutionary particle filter with the immune genetic algorithm

The effectiveness of the proposed algorithm can be observed from a practical tracking process. That is the estimation of single-variable non-stationary economic changes [28]. The movement model and observation model are shown as

$$x(k) = 0.5x\,(k-1) + \frac{25\,(k-1)}{1 + x\,(k-1)^2} + 8\cos\left(1.2\,(k-1)\right) + \omega(k) \tag{19}$$

and

$$y(k) = \frac{x(k)^2}{20} + v(k), \tag{20}$$

where $\omega(t)$ and $v(t)$ are both Gaussian noise with zero means and unit variances. The particle number applied is 100. In order to reduce the running time, a threshold for generations of evolution of immune optimization is introduced. If the effective particle number is less than the threshold value, the immune optimization algorithm begins. The threshold can take different values depending on the circumstances. Fig. 2 shows the result estimated by the standard particle filter and the immune optimization particle filter, respectively. It can be found in Fig. 2 that the particle filter with IGA has smaller error than the standard particle filter when the state jumps abruptly, which can be seen from the rectangular box in the figure. This is because there are no particles near the true state after the state jumps and all the particles have zero measurement likelihood [19]. On the other hand, for the proposed algorithm with the advancement for adding crossover, mutation and other operators, the particles set with serious "sample impoverishment" characteristic generate many new samples. These new samples can obviously increase the number of meaningful particles after re-sampling, thus enhance the tracking performance of the particle filter with IGA. It can be concluded that the IGA can improve the particle filter's performance because of its improvement on particle diversity and effectiveness.

4. Target tracking in video surveillance scenario

4.1. Video sequences

The experiment includes two videos. One is a commonly used video and another is taken from the actual surveillance camera, in which a pre-selected moving target is aimed to be tracked. In the first video, there contains a remote control aircraft and the controller. The second video sequence contains two people walking in front of a stationary camera. The weighted color distribution of pixels inside the ellipse represents the moving target which we aim to track. The second video target's normalized color histogram of target feature can be seen in Fig. 3. We can use it to obtain the observation probability distribution of the target according to Eqs. (5)–(9). After that we can gain the importance weights of particles. The main difficulties of the used video sequences are: (1) The color of the chosen target feature is similar to the video background, (2) The color is also similar to a behind passing person and (3) When the passer-by is passing the target, there is a temporal near-complete occlusion.

4.2. Tracking robustness

We compare the particle filter with IGA with the standard particle filter from the aspect of tracking robustness in the experiment. The tracking results can be seen in Figs. 4 and 5. From the video in Fig. 4(a), it can be seen that the standard particle filter can successfully track the target at the beginning. But when the controller behind passes by and over the target, it cannot accurately distinguish between the target and the controller because of the similar color features of their appearance, then results in wrong tracking. In the video in Fig. 5(a), again because of the similar color features of the two persons' dress, the standard particle filter tracks the wrong person as well. In these cases, the particles of the standard particle filter are seriously affected by impoverishment and overlap most of the time. So when the passer-by causes occlusion, the particles may be trapped into some false local maximum, and cannot recover from the failure when the target reappears.

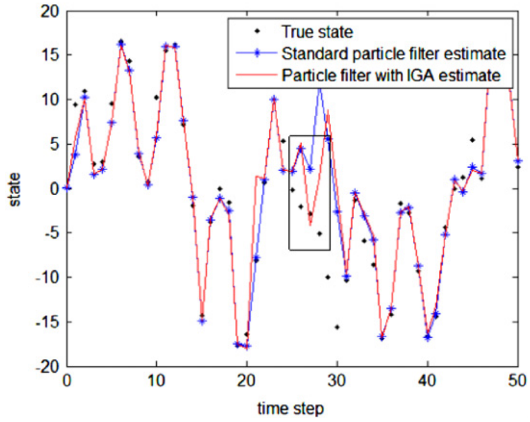


Fig. 2. The estimation of specific state by the standard particle filter and the particle filter with IGA, respectively.

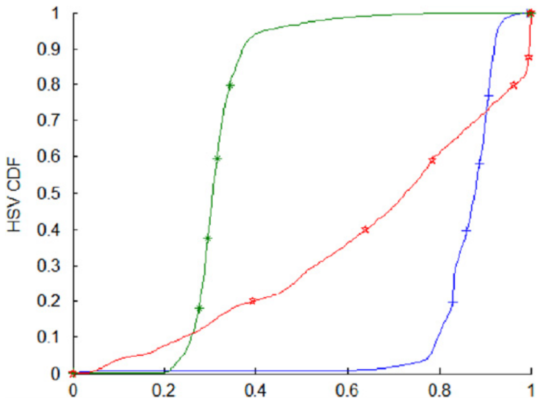


Fig. 3. The normalized color histogram of target feature.

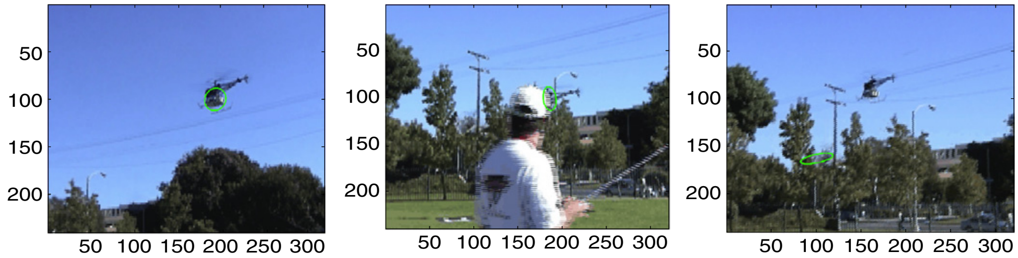
But on the contrary, in Figs. 4(b) and 5(b), the proposed particle filter with IGA can deal with the situation well and tracks successfully throughout the video (even if there exist color similarity and occlusion). Due to the adding of crossover and mutation, which can be seen in Section 3.2, the particles in the particle filter with IGA can exploit more wide areas to find suitable particles to approximate the target's true state during the occlusion. And IGA can maintain the diversity of the particle set. All of these aforementioned guarantee that the proposed filter can successfully re-track the target when it appears again.

4.3. Tracking efficiency

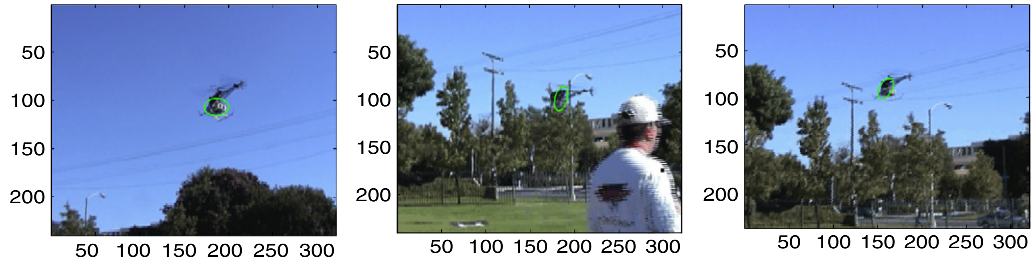
Tracking efficiency is another aspect which can show that the particle filter with IGA is superior to the standard particle filter. That is the number of meaningful particles as shown in Fig. 6. It can be seen that the difference of the meaningful particle number of the two algorithms in tracking the second video's target. From Fig. 6, in general, the meaningful particle number of the particle filter with IGA is greater than that of the standard particle filter. After inserting crossover, mutation and other operations, the proposed algorithm repeats selecting particles which can represent the target feature better in the new particle set. So after several cycles, the final particle set can express the state of moving target more accurately. And the more meaningful particles, the smaller sample impoverishment. This is another way to show that the proposed particle filter with IGA has better performance.

5. Conclusions

In this paper, we propose a new evolutionary particle filter with IGA for video tracking. The study of new algorithm focuses on the sample impoverishment brought by re-sampling. We add IGA before re-sampling and regard the particles as antibodies of the immune system. Through crossover and mutation process, the immune system produces a large number of new antibodies. Regulatory mechanisms of antibodies (particles), such as promotion and suppression, ensure the diversity of the particle set and increase the number of meaningful particles. From the estimation of standard verification model

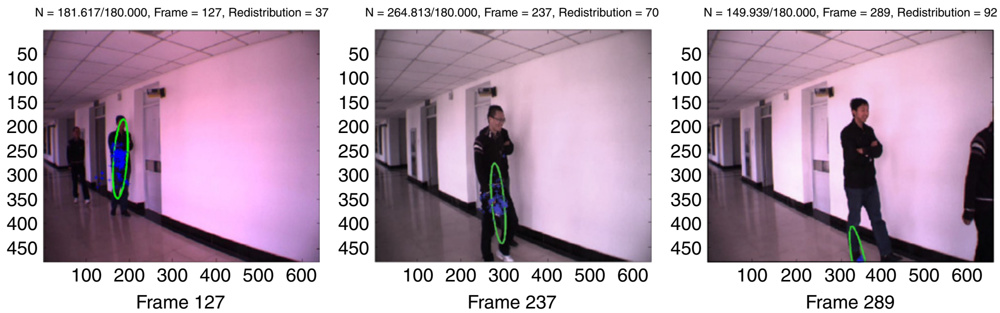


(a) The standard particle filter.

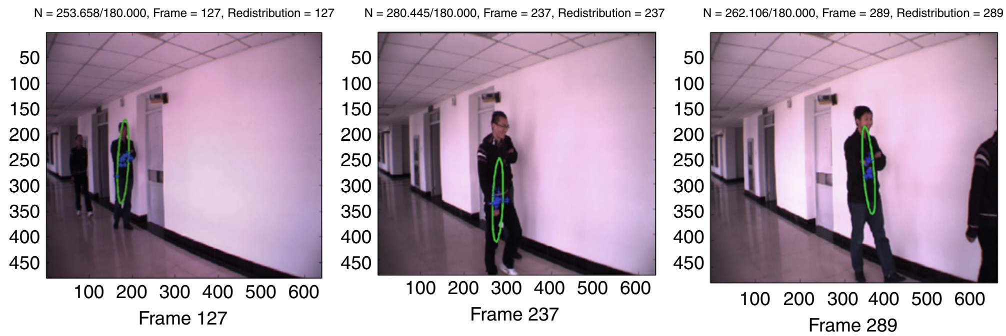


(b) The particle filter with IGA.

Fig. 4. Helicopter tracking by the standard particle filter and the particle filter with IGA, respectively.



(a) The standard particle filter.



(b) The particle filter with IGA.

Fig. 5. Person tracking by the standard particle filter and the particle filter with IGA, respectively.

and moving target tracking on complex background, we verify the proposed particle filter has better performance than the standard particle filter in the error of state estimation, error tracking of video, and the number of meaningful particles. It shows that the particle filter with IGA increases the number of meaningful particles, and makes the particle set express the true state better. It can increase the accuracy of state estimation and reduce the error.

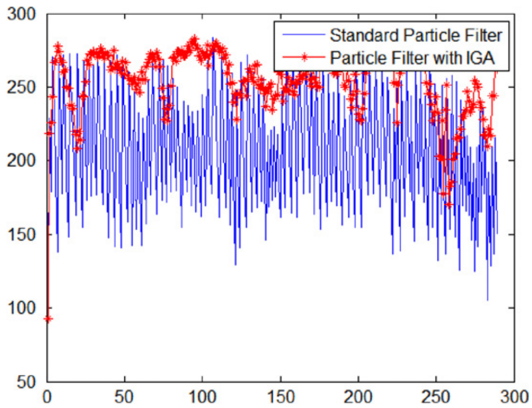


Fig. 6. Comparison of the number of meaningful particles of the two algorithms.

In future, we will extend the proposed particle filter with IGA to the multiple objects tracking problem with occlusions in complex and dynamic changing environment. In this situation, samples will be drastically impoverished and the proposed particle filter with IGA will become more important to increase particle diversity.

Acknowledgments

This work was supported in part by the National Nature Science Foundation of China (Nos. 60975059, 60775052), Specialized Research Fund for the Doctoral Program of Higher Education from the Ministry of Education of China (No. 20090075110002), and the Project of the Shanghai Committee of Science and Technology (Nos. 10JC1400200, 10DZ0506500, 09JC1400900).

References

[1] C.M. Huang, L.C. Fu, Multitarget visual tracking based effective surveillance with cooperation of multiple active cameras, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 41 (1) (2011) 234–247.
[2] S. Gidel, P. Checchin, C. Blanc, T. Chateau, L. Trassoudaine, Pedestrian detection and tracking in an urban environment using a multilayer laser scanner, IEEE Transactions on Intelligent Transportation Systems 11 (3) (2010) 579–588.
[3] M. Kristan, S. Kovačič, A. Leonardis, J. Perš, A two-stage dynamic model for visual tracking, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 40 (6) (2010) 1505–1520.
[4] J. Zhu, Y. Lao, Y.F. Zheng, Object tracking in structured environments for video surveillance applications, IEEE Transactions on Circuits and Systems for Video Technology 20 (2) (2010) 223–235.
[5] B. Han, Y. Zhu, D. Comaniciu, L.S. Davis, Visual tracking by continuous density propagation in sequential Bayesian filtering framework, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (5) (2009) 919–930.
[6] C. Shen, J. Kim, H. Wang, Generalized kernel-based visual tracking, IEEE Transactions on Circuits and Systems for Video Technology 20 (1) (2010) 119–130.
[7] P. Cui, L.-F. Sun, F. Wang, S.-Q. Yang, Contextual mixture tracking, IEEE Transactions on Multimedia 11 (2) (2009) 333–341.
[8] H. Weiming, Z. Xue, H. Min, S. Maybank, Occlusion reasoning for tracking multiple people, IEEE Transactions on Circuits and Systems for Video Technology 19 (1) (2009) 114–121.
[9] W.L. Lu, K. Okuma, J.J. Little, Tracking and recognizing actions of multiple hockey players using the boosted particle filter, Image and Vision Computing 27 (1–2) (2009) 189–205.
[10] A. Łoza, L. Mihaylova, D. Bull, N. Canagarajah, Structural similarity-based object tracking in multimodality surveillance videos, Machine Vision and Applications 20 (2) (2009) 71–83.
[11] M. Isard, A. Blake, CONDENSATION—conditional density propagation for visual tracking, International Journal of Computer Vision 29 (1) (1998) 5–28.
[12] N. Gordon, D. Salmond, C. Ewing, Bayesian state estimation for tracking and guidance using the bootstrap filter, Journal of Guidance, Control and Dynamics 18 (6) (1995) 1434–1443.
[13] J.S. Liu, R. Chen, Sequential Monte Carlo methods for dynamic systems, Journal of the American Statistical Association 93 (1998) 1032–1044.
[14] E. Maggio, A. Cavallaro, Accurate appearance-based Bayesian tracking for maneuvering targets, Computer Vision and Image Understanding 113 (4) (2009) 544–555.
[15] H. Medeiros, G. Holguín, P.J. Shin, J. Park, A parallel histogram-based particle filter for object tracking on SIMD-based smart cameras, Computer Vision and Image Understanding 114 (11) (2010) 1264–1272.
[16] X.-Y. Xu, B.-X. Li, Adaptive Rao–Blackwellized particle filter and its evaluation for tracking in surveillance, IEEE Transactions on Image Processing 16 (3) (2007) 838–849.
[17] R. Cabido, A.S. Montemayor, J.J. Pantrigo, B.R. Payne, Multiscale and local search methods for real time region tracking with particle filters: Local search driven by adaptive scale estimation on GPUs, Machine Vision and Applications 21 (1) (2009) 43–58.
[18] J. Fourie, S. Mills, R. Green, Harmony filter: a robust visual tracking system using the improved harmony search algorithm, Image and Vision Computing 28 (12) (2010) 1702–1716.
[19] S. Park, J.P. Hwang, E. Kim, H.-J. Kang, A new evolutionary particle filter for the prevention of sample impoverishment, IEEE Transactions on Evolutionary Computation 13 (4) (2009) 801–809.
[20] M.S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, IEEE Transactions on Signal Processing 50 (2) (2002) 174–188.
[21] K.S. Hariharakrishnan, Fast object tracking using adaptive block matching, IEEE Transactions on Multimedia 7 (5) (2005) 853–859.

[22] W. Qu, D. Schonfeld, Real-time distributed multi-object tracking using multiple interactive trackers and a magnetic-inertia potential model, IEEE Transactions on Multimedia 9 (3) (2007) 511–519.
[23] J. Fang, J. Ali, Realization of an autonomous integrated suite of strapdown astro-inertial navigation systems using unscented particle filtering, Computers and Mathematics with Applications 57 (2009) 169–183.
[24] D. Comaniciu, V. Ramesh, P. Meer, Kernel-based object tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (5) (2003) 564–575.
[25] K. Nummiaro, E. Koller-Meier, L. Van Gool, An adaptive color-based particle filter, Image and Vision Computing 21 (1) (2003) 99–110.
[26] J.H. Park, G.S. Lee, S.Y. Park, Color image segmentation using adaptive mean shift and statistical model-based methods, Computers and Mathematics with Applications 57 (2009) 970–980.
[27] Y.-S. Ding, B. Liu, L.-H. Ren, K.-R. Hao, Intelligent Control and Optimization Based on Bio-Network, Science Press, Beijing, 2010.
[28] N. Gordon, D.J. Salmond, A.F.M. Smith, Novel approach to nonlinear/non-Gaussian Bayesian state estimation, IEE Proceedings, Part F: Radar and Signal Processing 140 (2) (1993) 107–113.